

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
APPLICATION FOR PATENT

TITLE: A METHOD TO ASSOCIATE INPUT AND OUTPUT INTERFACES WITH
PACKETS READ FROM A MIRROR PORT

INVENTOR: Peter Phaal

BACKGROUND OF THE INVENTION

The present invention relates to network monitoring systems. More particularly, the present invention relates to an apparatus and method to analyze the performance of a network switch by monitoring port interfaces of the switch.

In a switched network, the problems and solutions of a shared network do not always apply. Each node may have a dedicated switched port, so bad packets from that node do not affect the rest of the nodes on the network. This feature is somewhat offset by the fact that troubleshooting switched networks is more difficult and requires more specialized tools. No longer can a conventional local area network (LAN) analyzer see all packets and determine which node is the cause of the problem. For this reason, measuring overall traffic levels and studying individual frame statistics is not straightforward in a switched environment. Several switch vendors have proposed ways to remedy this problem with innovative techniques, such as internal RMON data collection and switch port aliasing.

A packet is only forwarded to one port on the switch, so a LAN analyzer connected to one port of a switch will not, by definition, catch all packets flowing through the switch. Many switch vendors have attempted to solve the traffic-monitoring problem by designing statistic collecting modules into the switch. These modules are either based on RMON statistics or a proprietary collection mechanism. For instance, some commercially available network switches keep RMON and SNMP statistics on each packet and can report overall switch traffic levels to any SNMP agent or any proprietary management software.

These modules can diagnose traffic-level problems on a switch with the previously mentioned techniques. Unlike shared networks, however, switched networks do not suffer from throughput degradation due to faster nodes and too many nodes, so these may not

EK381340040US

cause problems on a switched network. In a workgroup switch, each new user receives a certain amount of dedicated bandwidth. As long as the high bandwidth connection to the switch can handle the traffic, no traffic-level problems should occur.

Individual frame analysis on a switch provides an interesting proposition. Packets are not forwarded to all ports on a switch, so there is no logical place to plug in a LAN and view all packets. Switch vendors have designed many ways to overcome this troubleshooting drawback. Two ways are port aliasing and adding a repeater to monitor a port.

Port aliasing, also called port mirroring or port spanning, is a mechanism by which a switch monitors all traffic flowing through one or any combination of selected ports and mirrors the network traffic to a special alias port. In additional, in some designs, only a portion of the network traffic of the selected port(s) is mirrored to the alias port for analysis. This allows a LAN administrator to examiner switch traffic one port at a time. If problematic packets are coming from a switch, they can usually be isolated in this fashion. Figure 1 shows how port aliasing works.

Even if a switch does not have advanced troubleshooting features, such as statistics-gathering modules and port aliasing, a few tricks can still be used to troubleshoot problematic switched networks. Adding a repeater is one such method. Adding a repeater refers to cascading a suspected problem node through a repeater into a switch, as shown in Figure 2. A port and a LAN analyzer are both connected to the repeater, and the repeater is, in turn, connected to the switch port. In this fashion, the analyzer can monitor any port on a switch for bad frames and other error conditions.

Although adding a repeater is a cheap and simple way to troubleshoot a switched network, it does have its drawbacks. When a repeater is inserted between the end node and the switched port, it is effectively changing the characteristics of the connection. For instance, collisions may now occur on the wire, whereas before they did not. If the switch-to-client connection was full-duplex before, it will become half-duplex after adding a repeater. These subtle changes in the connection could change the dynamics of the problem the system is trying to troubleshoot. There have also been some recent developments in using optical links to passively tap the switched port so that the basic characteristics of the connection are not changed.

However, unfortunately, the data produced by the protocol analyzer does not accurately reflect the traffic patterns with the switch, especially if more than one port is being

5

SUMMARY OF THE INVENTION

It is therefore an object of the present invention to provide a method and apparatus to monitor the network traffic of a network switch.

It is another object of the present invention to monitor the network traffic of a
5 network switch using a mirror port of the network switch.

It is yet another object of the present invention to reconstruct the network traffic using the mirror port of the network switch by analyzing the information that could have been provided by an embedded traffic monitor

The network monitoring system according to the present invention comprises a
10 monitor device coupling to a network switch through a mirror port interface of the network switch. The network switch supports a number of regular port interfaces connected to different network components and the mirror port connected to the monitor device. The mirror port mirrors all or a selected portion of the network traffic of any selected regular port interface(s) of the network switch. When the monitor device receives a data packet from the network switch,
15 the monitor device extracts the network address information such as the source and destination address from the data packet. The monitor device then examines internal lookup tables to determine whether there are corresponding physical port interfaces for the network address information. If the corresponding physical port interfaces for the network address information are not available in the lookup table, the monitor device interrogates the network switch for the corresponding physical port interfaces. The data packet is placed in a first-in-first-out buffer temporarily until the network switch responses to the monitor device's request. Finally, the
20 monitor device uses the physical port interfaces to perform further network analysis functions.

Additional objectives, features and advantages of various aspects of the present invention will become apparent from the following description of its preferred embodiments,
25 which description should be taken in conjunction with the accompanying drawings.

DETAILED DESCRIPTIONS OF THE PREFERRED EMBODIMENT

Figure 3 shows the basic elements of a network monitoring system 300 according to the present invention. The network monitoring system 300 is designed to monitor the performance of a network switch 310 by collecting statistical information of the port interface(s) of the switch 310. In the present example, one of the port interfaces of the switch is a mirror port 315 that is able to mirror at least some network traffic flowing through any selected ports. According to the present invention, a monitor device 335 can report switch traffic levels of one or any combination of the port interfaces 320 a,b...m of the switch 310 and therefore can diagnose traffic-level problems on the switch 310. For example, the mirror port 315 can mirror only one selected port interface, or mirror a combination of more than one port interfaces of the network switch. In the example as shown in Figure 3, two computers, Host A 325 and Host B 330, are connected to two port interfaces on the switch 310. In this preferred embodiment, the monitor device 335 is further connected to one of the switch port interfaces for communicating with the network switch 310.

According to the present invention, the monitor device 335 is coupled to the mirror port 315 of the network switch 310 so that all network traffic passing across the mirror port 310 will be forwarded to the monitor device 335. Therefore, the monitor device 335 can monitor and provide information about the network traffic passing through any port interface(s) of the network switch 310. This information may include statistics characterizing traffic load of any of the port interfaces 320 a,b...m.

In one preferred embodiment of the present invention, the monitor device 335 maintains lookup tables that map network addresses to the port interface indexes and the physical port interfaces. Each network address embedded in the data packet can be looked up in the table so that the associated physical port interface can be identified. By providing the monitor device 335 the physical port interface information for network address contained in each data packet passing through the mirror port 315, the monitor device 335 is able to examine and accurately analyze the packet flows through the examining port(s).

As discussed above, the monitor device 335 according to the present invention maintains internal lookup tables that associate the network addresses with the physical port interface that the data packets are received from. These internal lookup tables are continuously updated by learning from the network traffic. For example, when a data packet is received on

the mirror port interface 315, the source and destination address of that data packet is recorded and associated with the corresponding port interfaces. One of the features of the present invention is the method to propagate these tables.

Figure 4 shows the essential elements of a sample data packet 400 passing through a network switch. As shown in the figure, a data packet 400 contains a source address 410, a destination address 420, and other data 430. The source address 410 and the destination address 420 denote the network addresses of the sender and the receiver respectively. Conventionally, each node has a unique network address. The first three bytes of the network address are called the block ID or organizationally unique identifier (OUI) and identify the manufacturer of the equipment. The IEEE assigns them. The other three bytes are called the device ID and are assigned to each device by each manufacturer. These numbers are always unique.

Figure 5 is an example illustrating the exchanging of a data packet between the monitor device 510 and the network switch 515 according to the present invention. The monitor device 510 is connected to the network switch 515 through the mirror port. In this example, the mirror port interface mirrors the network traffic of the port interface connecting Host B and the network switch. Therefore, the data packet sent from Host A to Host B is mirrored and forwarded to the monitor device through the mirror port. In this embodiment, the network switch 515 supports Simple Network Management Protocol ("SNMP") and is able to process messages relating to the Bridge MIB (Management Information Base) as defined by RFC 1493. However, the present invention also supports other forms of communication between the monitor device and the network switch. For example, the monitor device can communicate with the network switch using any commercially available or propriety communication protocol.

In this preferred embodiment according to the present invention, the monitor device performs the following steps when the data packet flows from Host A to Host B is mirrored on the mirror port interface:

1. Host A generates a data packet containing the source address of Host A and the destination address of Host B. The data packet is sent to the network switch.
2. The switch examines the destination address and forwards the data packet to the port interface connected to Host B.

3. The switch copies the data packet and sends a copy of the data packet to the monitor device through the mirror interface.

4. The monitor device examines the data packet and extracts the source address and destination address embedded in the data packet. When the lookup tables in the monitor device do not contain an entry for any of the source and destination addresses, the monitor device issues an SNMP request to the network switch to obtain the logical port index associated with the network address.

5. The switch sends an SNMP response containing the port index associated with the network address back to the monitor device.

6. Using the port index information, the monitor device issues a second SNMP request to the network switch to inquire the physical port interface (i.e. an index that is locally unique to that port interface) associated with the logical port index.

7. The switch returns an SNMP response containing information about the physical port interface associated with the logical port index to the monitor device.

After receiving the physical port interface information, the monitor device can either update the lookup table or create new entry for the lookup table.

Figures 6, 7 and 8 are flow charts describing the algorithm of how the monitor device maps network addresses to switch interfaces. In the interests of clarifying the algorithm, it is described in terms of identifying the switch interface that data packets are received on (i.e. the switch interface associated with the source address). The algorithm should equally be applied to the destination address to identify the switch interface that packets are sent on.

According to the present invention, the monitor device maintains two internal lookup tables to assist the translation: (1) Address Table ("addrTbl") for mapping a network address to a port index; and (2) Port Table ("prtTbl") for mapping a port index to a physical port interface. If an entry exists in these two tables, the monitor maps the network address to the corresponding physical switch interfaces and the mapping function is then complete. Otherwise, an entry for the data packet is put into a first-in-first-out buffer ("FIFO") for a short period of time to allow the monitor device to interrogate the network switch and create a new mapping entry. When the packet is removed from the FIFO, the entry corresponding the source and destination addresses should now exist and the interface information can be associated with the packet.

Figure 6 is a flow chart showing the sequence of events taken place when the monitor receives a packet from the switch mirror interface.

In Step 610, the monitor device waits to receive a data packet passing across the mirror port. When the monitor device receives a data packet from the mirror port, the data packet is time stamped and the source address is extracted from the data packet. The monitor then checks the address tables `addrTbl` to see whether there is an entry for the source address (Step 620). If the address table `addrTbl` does not contain an entry for the source address, a new entry for the source address is created (Step 630). If a new entry is needed or an existing entry is stale and needs refreshing, the monitor device then interrogates the network switch for the corresponding port interface index to the source address. The monitor device issues an SNMP request to the network switch to get the port interface index associated with the source address. The monitor device sends an `SNMP_GET` request to the switch to map the source address to the corresponding port index. (i.e. `SNMP_GET .1.3.6.1.2.1.17.4.3.1.2.<a1>.<a2>.<a3>.<a4>.<a5>.<a6>`, where `a1...a6` are the 6 bytes of an address will return an integer index for the associated port. The time of the SNMP request is then noted in this step (Step 640). In Step 650, when the switch responds to the request with the corresponding port index for the source address, the data packet is added to the first-in-first-out buffer FIFO (see Figure). The process is then looped back to Step 610. On the other hand, when the address table `addrTbl` contains an entry for the source address, a valid existing entry was obtained from the address table `addrTbl`. The entry in the port table `prtTbl` is used to obtain the switch interface for this source address in Step 660. After the entry is obtained from the table `prtTbl`, the mapping algorithm is completed and the packet can be handed off for analysis in Step 670.

Figure 7 is a flow chart showing the sequence of events taken place when the monitor device receives an SNMP response from the switch.

In Step 710, the monitor device waits for an SNMP response. If the response is for responding to a previous network address to port index mapping request, then the corresponding port interface index is provided to the monitor device. The corresponding entry in the address table `addrTbl` is updated with the port index from the SNMP response (Step 720). The entry in the port table `prtTbl` associated with the port index is identified in Step 730. If an entry does not exist for this port, an entry is created (Step 740). If the entry is new, or an existing

entry is stale and needs refreshing, an SNMP request is made to find the switch interface associated with the port index. The monitor device then sends another SNMP_GET request to the switch to map the port index to the corresponding physical port interface. (i.e. SNMP_GET .1.3.6.1.2.1.17.1.4.1.1.<i>, where i is the index returned in the previous query gives the physical port interface associated with the index (Step 750). The monitor device then determines whether a new request can be issued. If the entry is stale and no requests have been made within a specified interval (minAttempt), then a new request is allowed (Step 760) If the request is to a switch port to switch interface request, then the monitor device updates the entry in the port table prtTbl associated with the switch port in Step 770.

Figure 8 is a flow chart showing how the monitor device removes the data packet entry from the FIFO.

In Step 810, the monitor device waits for a short time interval (fifo_polling_interval). In Step 820, if there is a packet in the FIFO and it has been there for the maximum period set by the parameter maxDelay then remove it from the FIFO. The monitor device then finds the entry in the address table corresponding to the source address in the packet. In Step 830, if the mapping obtained from the address table is stale, the monitor device lookups the entry in the port table prtTbl corresponding to the port interface obtained from the address table addrTbl. In Step 840, if there is a valid entry in the port table prtTbl, then the associated physical port interface is obtained. In Step 850, if the interface is valid, then the entry in the address table addrTbl is updated. In step 860, if there is a valid entry in the address table, or if the corresponding entry is just updated, the entry is used to obtain the port interface index for the source address. Finally, in Step 870, when the corresponding physical port interface for the network address is obtained, the data packet off for analysis – even if the source interface of the source address is not successfully mapped. However, the data packet is less useful for network traffic analysis because of lack of physical port interface information.

In another preferred embodiment according to the present invention, the similar technique is applied to a network switch performing routing functions. In this embodiment, the monitor device monitors the network traffic of the routing switch by associating routing information with data packets obtained from the mirror port of the routing switch. The routing switch has an internal routing table containing the routing information relating to destination networks. The monitor device can obtain the information regarding the routing table from the

routing switch through SNMP requests (using the MIB-II specification described in RFC 1213). In this embodiment, the monitor device downloads the routing table into its internal memory and then makes queries against the routing table that maps a network address to a subnet and subnet mask, switch interface, next hop router and Autonomous System (AS) number path in the case of routers using Border Gateway Protocol (BGP) routing protocol. According to this preferred embodiment, the monitor device periodically updates the routing table by copying the routing table from the routing switch using SNMP requests (or any other commercially available/propriety communication protocol).

In another embodiment according to the present invention, the routing switch is designed to respond to inquiry for routing and port interface information relating to a particular network address. In this case, the monitor device is not required to maintain a duplicated copy of the routing table because the routing switch performs the actual routing table lookup upon each request from the monitor device. This embodiment is similar to the method as discussed with Figures 6, 7, and 8. In this embodiment, the requested routing information is obtained from the routing tables located in the routing switch.

The foregoing description has been limited to a specific embodiment of this invention. It will be apparent, however, that variations and modifications may be made to the invention, with the attainment of some or all of the advantages of the invention. Therefore, it is the object of the appended claims to cover all such variations and modifications as come within the true spirit and scope of the invention.